

### **AMENDMENTS TO THE CLAIMS**

Applicant submits below a complete listing of the current claims, including marked-up claims with insertions indicated by underlining and deletions indicated by strikeouts and/or double bracketing. This listing of claims replaces all prior versions, and listings, of claims in the application:

#### **Listing of the Claims**

1. (Currently amended) Apparatus adapted for use in a peer-to-peer collaboration system, the apparatus comprising a computer system with a memory and a computer-readable medium having computer executable modules, the computer-executable modules comprising:

an activity program adapted to implement a portion of a collaboration session, the activity program maintaining a local data copy of a shared space in response to user actions within the collaboration system, and the activity program generating a component update request in response to an action by a user within the collaboration session;

a component manager that receives the component update request from the activity program and has a parser that extracts from the request URL information which identifies the location of a file containing software component resources for satisfying the component update request;

a download manager that receives the URL information from the component manager and has a file retriever which asynchronously retrieves the file from the specified location and places the file in a staging area in the memory; and

an install manager that asynchronously installs the file,  
wherein the component manager is adapted to determine whether the requested software component is already installed on the computer system and to selectively invoke the download manager based on the determination.

2. (Original) The apparatus of claim 1 wherein the file contains an OSD description of the software component resources.

3. (Original) The apparatus of claim 1 wherein the component manager comprises a security section that validates the file before installation.

4. (Original) The apparatus of claim 1 further comprising a manifest which contains a list of all software components installed on the computer system.

5. (Original) The apparatus of claim 4 wherein the component manager comprises a mechanism that responds to the request by checking the manifest to ascertain whether the requested software component is already installed on the computer system.

6. (Previously presented) The apparatus of claim 1 wherein the component manager comprises a polling mechanism that periodically polls component locations to locate new component versions.

7. (Previously presented) The apparatus of claim 1 wherein the software component may be a system component that is required for operation of the apparatus or an application component that is not required for operation of the apparatus and wherein the apparatus further comprises a system component manager that receives a request for the system component and a system component installer that is started by the system component manager.

8. (Previously presented) The apparatus of claim 1 wherein the activity program generates the update request in response to receiving an invitation for a user to join the collaboration session, the invitation being generated in response to an action by a user within the collaboration session.

9. (Previously presented) The apparatus of claim 1 wherein the activity program generates the update request in response to receiving an update delta for the collaboration session, the update delta being generated in response to an action by a user within the collaboration session.

10. (Original) The apparatus of claim 1 wherein the component manager comprises an activation factory for activating installed software components.

11. (Currently amended) A method of operating a computer system with a memory as part of a peer-to-peer collaboration system comprising at least one other computer system maintaining a first local data copy of a shared space with a memory, the method comprising:

(a) generating a component update request in response to receiving information about a component being used in a collaboration session involving the at least one other computer, the component update request identifying the component;

(b) parsing the request to extract from the request URL information which identifies the location of a file containing software component resources for satisfying the request and an identification of the component;

(c) determining, based on the identification of the component, availability of the component;

(d) selectively in response to the determination, using the URL information to asynchronously retrieve the file from the specified location; and

(e) asynchronously installing the component from the file, whereby the computer system can use the component to maintain a second local data copy of the shared space that is synchronized with the first local data copy.

12. (Original) The method of claim 11 wherein the file contains an OSD description of the software component resources.

13. (Previously presented) The method of claim 11 further comprising:

(f) validating the file before installation.

14. (Previously presented) The method of claim 11 further comprising:

(f) operating the computer system to implement a portion of the collaboration session using the component.

15. (Previously presented) The method of claim 11 wherein step (c) comprises checking a manifest to ascertain whether the requested software component is already installed on the computer system before retrieving the file.

16. (Previously presented) The method of claim 11 further comprising:

(f) periodically polling component locations to locate new component versions.

17. (Previously presented) The method of claim 11 wherein the component may be a system component that is required for operation of the apparatus or an application component that is not required for operation of the apparatus and wherein the method further comprises:

(f) when the component is a system component, installing the component with a separate system component manager, which receives a request for a system component, and a separate system component installer that is started by the system component manager.

18. (Previously presented) The method of claim 17 wherein step (f) comprises shutting the system component manager down before installing in-use components.

19. (Previously presented) The method of claim 18 wherein step (f) further comprises restarting the system component manager after the system component has been installed.

20. (Previously presented) The method of claim 11 further comprising:

(f) activating the installed software component with an activation factory.

21. (Previously presented) A computer program product for use in a peer-to-peer collaboration system including a computer system with a memory, the computer system being adapted to receive an update request generated as a result of user interaction with the peer-to-peer collaboration system, the computer program product comprising a computer usable medium having computer readable program code thereon, including:

program code that extracts from the request information which identifies the location of a first file for satisfying the request;

program code that uses the information to retrieve the first file;

program code that extracts from the first file an indicia of a trusted supplier and obtains second location information of a second file, the second file containing a second component;  
program code that uses the second location information to retrieve the second file;  
program code to extract from the second file the second component and an indicia of a supplier of the second component; and  
program code that selectively installs the second component when the indicia of the supplier is consistent with the indicia of a trusted supplier.

22. (Previously presented) The computer program product of claim 21 wherein the first file contains an OSD description of the software component , including dependent components.

23. (Original) The computer program product of claim 21 wherein the program code that retrieves the second file is the same program code that retrieves the first file.

24. (Previously presented) The computer program product of claim 21 wherein the program code that extracts from the first file the indicia of a trusted supplier extracts a fingerprint of a trusted supplier.

25. (Previously presented) The computer program product of claim 21 wherein the program code that uses the information to retrieve the file from the specified location comprises program code that checks a manifest to ascertain whether the requested software component is already installed on the computer system before retrieving the file.

26. (Previously presented) The computer program product of claim 21 wherein the program code that selectively installs the second component selectively prompts a user for authorization to install the second component when the indicia of the supplier is not consistent with the indicia of a trusted supplier.

27. (Previously presented) The computer program product of claim 21 wherein the software component may be a system component that is required for operation of the apparatus

or an application component that is not required for operation of the apparatus and wherein the computer program product further comprises program code that installs a system component with a separate system component manager that receives a request for the system component and a separate system component installer that is started by the system component manager.

28. (Original) The computer program product of claim 27 wherein the program code that installs system components with a separate system component manager and a separate system component installer comprises program code that shuts the system component manager down before installing in-use components.

29. (Previously presented) The computer program product of claim 28 wherein the program code that installs the system component with a separate system component manager and a separate system component installer further comprises program code that restarts the system component manager after the system component has been installed.

30. (Original) The computer program product of claim 21 further comprising program code that activates installed software components with an activation factory.

31. (Previously presented) A computer data signal embodied in a carrier wave for use in a peer-to-peer collaboration system including a computer system with a memory, the computer system being adapted to receive an update request generated as a result of user interaction with the peer-to-peer collaboration system, the computer data signal comprising:

- program code that extracts from the request URL information which identifies the location of a first file containing software component resources for satisfying the request;

- program code that uses the URL information to asynchronously retrieve the first file from the specified location;

- program code that extracts from the first file dependency information identifying a second component upon which the first component is dependent;

- program code that selectively downloads a second file when the dependency information indicates the second component is in the second file; and

- program code that installs the first component and the second component.

32. (Currently amended) Apparatus for use in a peer-to-peer collaboration system comprising a computer system with a memory and at least one other computer system maintaining a first local data copy of a shared space, the apparatus comprising:

means for implementing a collaboration session for a user, the means for implementing adapted to receive an indication of a component in use within the collaboration session involving the at least one other computer and to selectively generate an update request for the component;

means responsive to the request, for parsing the request to extract from the request URL information which identifies the location of a file containing software component resources for satisfying the request;

means for receiving the URL information and asynchronously retrieving the file from the identified location; and

means cooperating with the parsing means for asynchronously installing the component from the file, whereby the computer system can use the component to maintain a second local data copy of the shared space that is synchronized with the first local data copy.

33. (Original) The apparatus of claim 32 wherein the file contains an OSD description of the software component resources.

34. (Original) The apparatus of claim 32 wherein the parsing means comprises means for validating the file before installation.

35. (Original) The apparatus of claim 32 further comprising a manifest which contains a list of all software components installed on the computer system.

36. (Original) The apparatus of claim 35 wherein the parsing means comprises means responsive to the request for checking the manifest to ascertain whether the requested software component is already installed on the computer system.

37. (Original) The apparatus of claim 32 wherein the parsing means comprises means for periodically polling component locations to locate new component versions.

38. (Previously presented) The apparatus of claim 32 wherein the software component may be a system component that is required for operation of the apparatus or an application component that is not required for operation of the apparatus and wherein the apparatus further comprises a system component manager that receives a request for the system component and a system component installer that is started by the system component manager.

39. (Previously presented) The apparatus of claim 38 wherein the system component installer comprises means for shutting the system component manager down before installing an in-use component.

40. (Previously presented) The apparatus of claim 39 wherein the system controller comprises means for restarting the system component manager after system component has been installed.

41. (Original) The apparatus of claim 32 wherein the parsing means comprises means for activating installed software components.